

CRACKING
THE CASE WITH

FLEXBOX



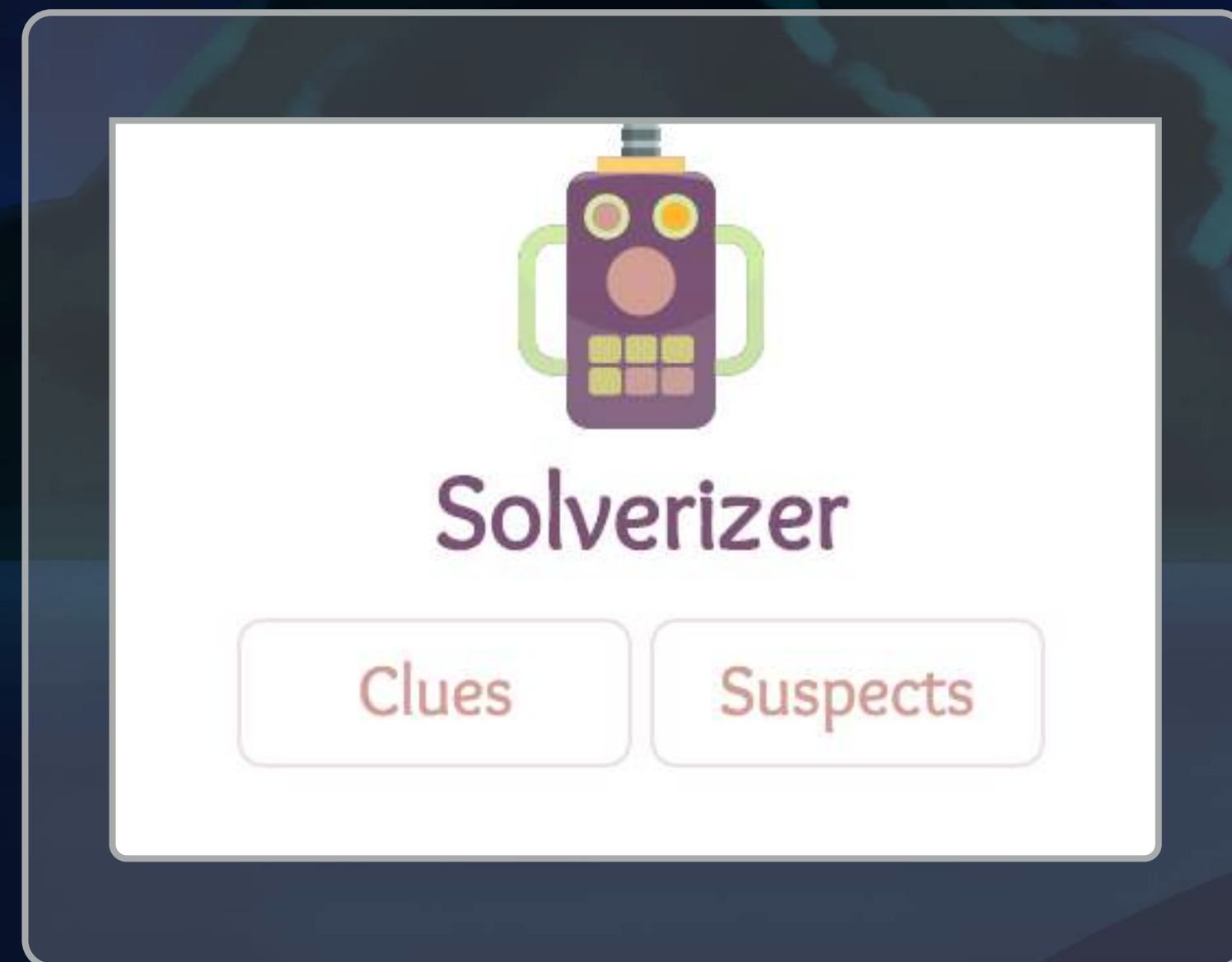


Level 2 – Section 1

Justification and Order

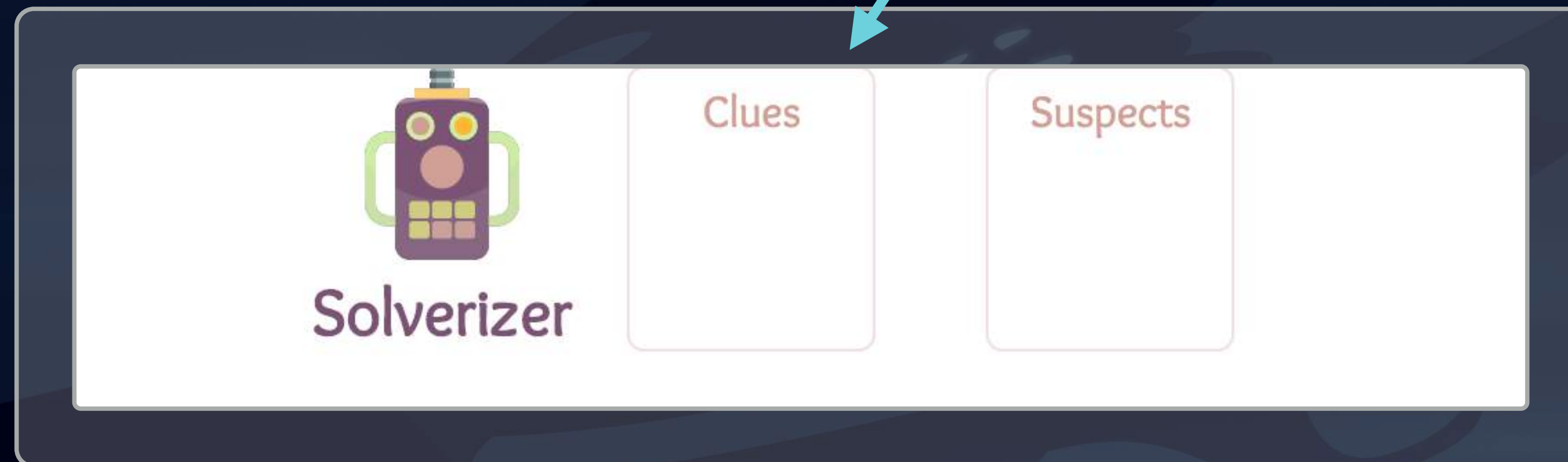
Distributing Space Along Flex Lines

Example Patterns



Positioning elements vertically

Positioning elements horizontally



Flex Items Are Distributed Along Axes

The main axis is determined by the `flex-direction`.

index.html

HTML

```
<nav>
  <a ...>...</a>
  <a ...>...</a>
  <a ...>...</a>
</nav>
```

site.css

CSS

```
@media screen and (min-width: 375px) {
  nav {
    display: flex; } }
```

start

center

end



Solverizer

Clues

Suspects

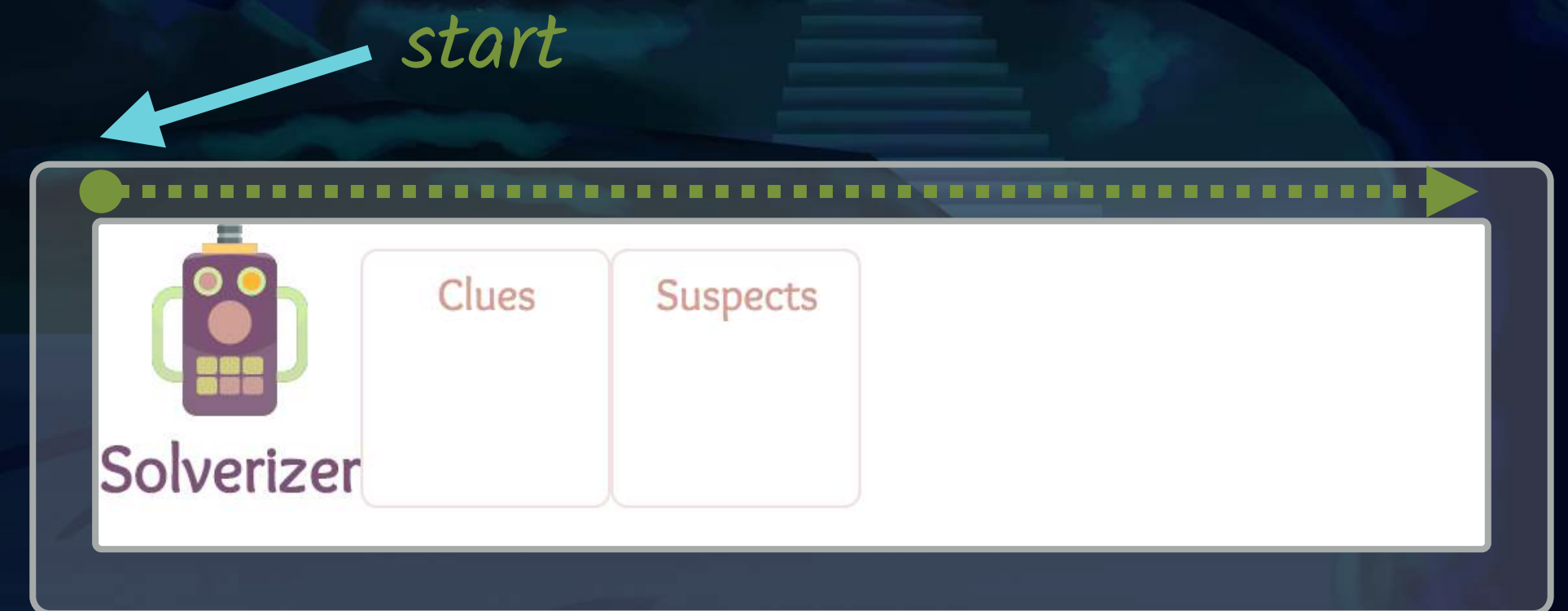
justify-content

Used to distribute space on the main axis, the default value is **flex-start** and it accepts:
flex-start | **flex-end** | **center** | **space-between** | **space-around**.

site.css

CSS

```
@media screen and (min-width: 375px) {  
  nav {  
    display: flex;  
    justify-content: flex-start; } }
```



site.css

CSS

```
@media screen and (min-width: 375px) {  
  nav {  
    display: flex;  
    justify-content: flex-end; } }
```



Spacing Values for justify-content

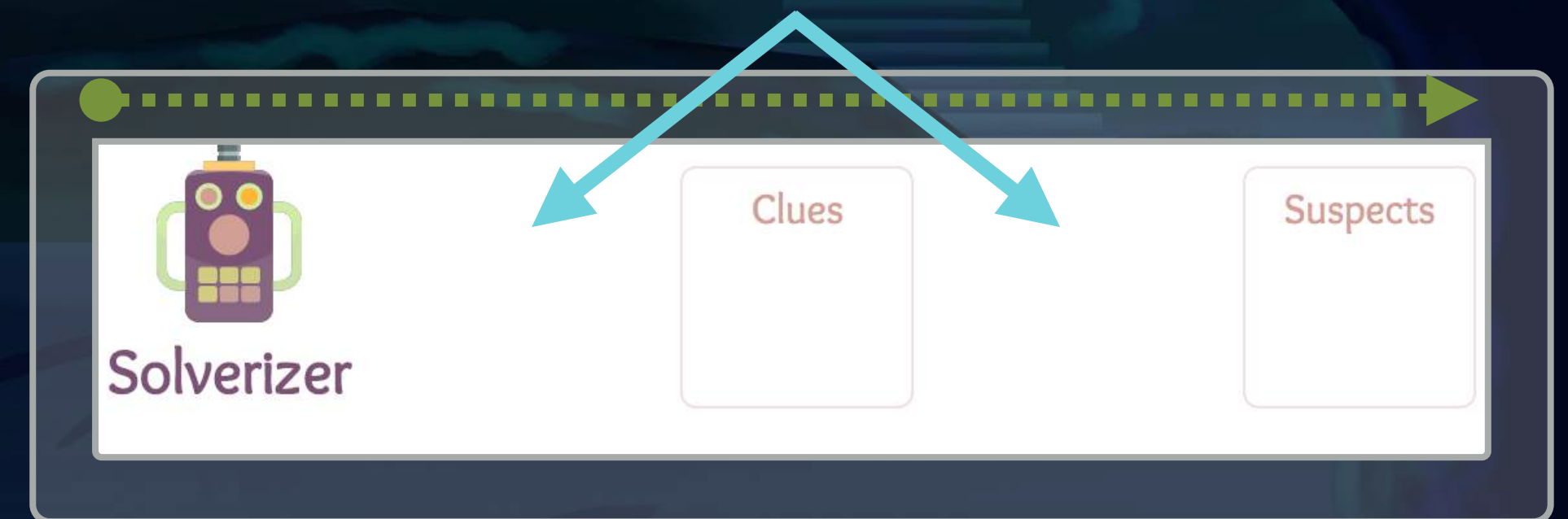
Flexbox's spacing algorithm distributes available space between elements.

site.css

CSS

```
@media screen and (min-width: 375px) {  
  nav {  
    display: flex;  
    justify-content: space-between; } }
```

Flush to edges, space in between

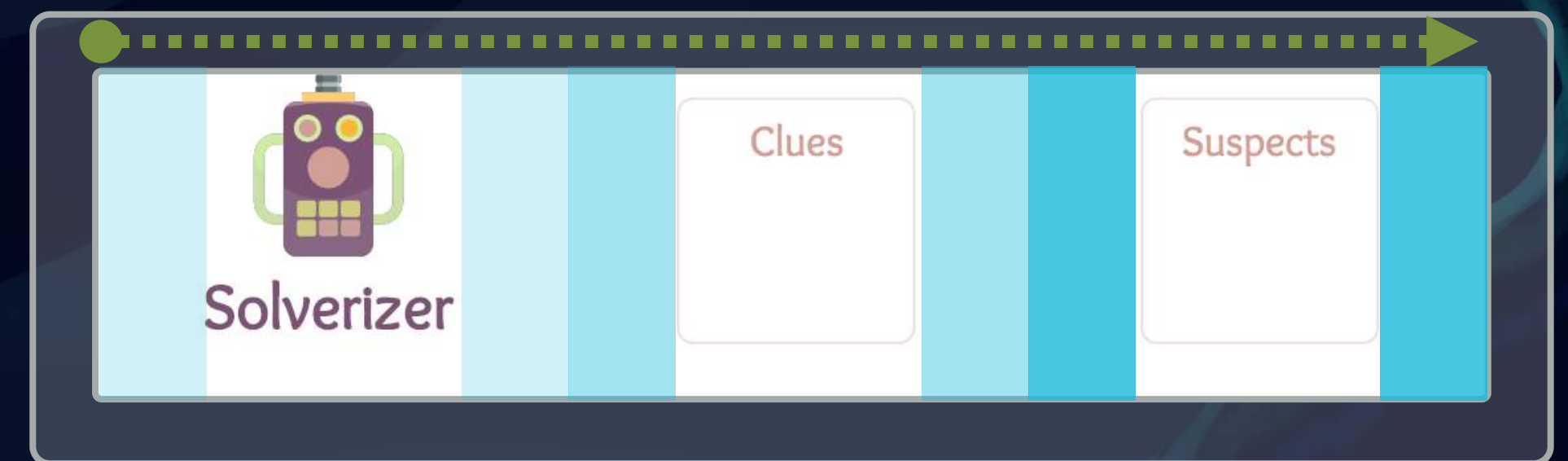


site.css

CSS

```
@media screen and (min-width: 375px) {  
  nav {  
    display: flex;  
    justify-content: space-around; } }
```

Space added around each element



Our Example Uses center

index.html

HTML

```
<nav>
  <a ...>...</a>
  <a ...>...</a>
  <a ...>...</a>
</nav>
```

site.css

CSS

```
@media screen and (min-width: 375px) {
  nav {
    display: flex;
    justify-content: center; } }
```

Content center aligns with container center



Adding Space With Non-flexbox CSS

Margin and padding can be applied to flex items.

index.html

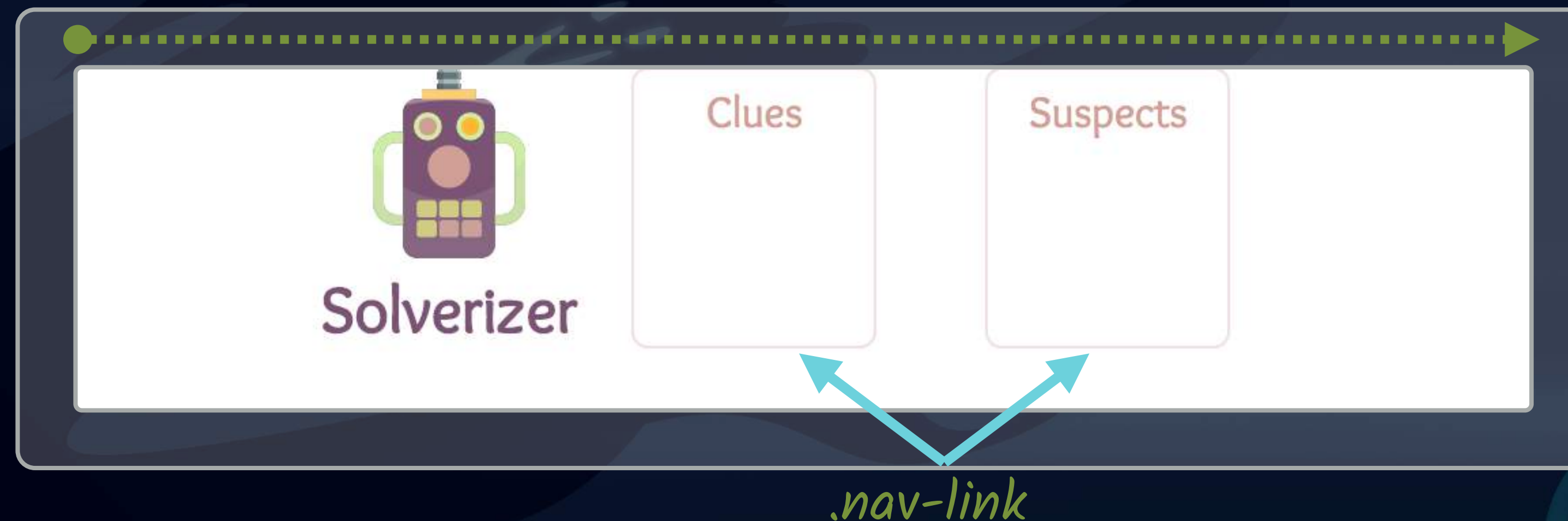
HTML

```
<nav>
  <a class="nav-brand">...</a>
  <a class="nav-link">...</a>
  <a class="nav-link">...</a>
</nav>
```

site.css

CSS

```
@media screen and (min-width: 375px) {
  nav {
    display: flex;
    justify-content: center; }
  .nav-link {
    margin: 0 20px;} }
```



Justifying Content in a Column

Main elements for layout are flex items with the `column` direction.

index.html

HTML

```
<main>
  ...
</main>
```

site.css

CSS

```
main {
  display: flex;
  flex-direction: column;
  justify-content: center; }
```

start

center of column

end



Villainr

Meet diabolical co-conspirators



CRACKING
THE CASE WITH

FLEXBOX

